LMI conditions for robust invariance of the convex hull of ellipsoids with application to nonlinear state feedback control

Hoai Nam Nguyen

LMI Conditions for Robust Invariance of the Convex Hull of Ellipsoids with Application to Nonlinear State Feedback Control

Hoai-Nam Nguyen†

***Abstract*—The convex hull of ellipsoids was suggested in the literature for robust invariance of constrained uncertain and/or time-varying linear discrete-time systems. It was show that a robust invariant set obtained with the convex hull of ellipsoids can be significantly larger than that with the ellipsoidal set. However, the design conditions are given in terms of bilinear matrix inequalities (BMIs), which are non-convex. The main purpose of this paper is to present a way to overcome this weakness by providing new convex linear matrix inequality (LMI) design conditions. It is shown that the conditions are losslessly extended to robust controlled invariance and to nonlinear state feedback control design. Two examples are included with comparison to earlier solutions from the literature to illustrate the results.**

I. Introduction

Stability analysis and stabilization against uncertain time-varying parameters of the plant with/without input and state constraints has been a challenging research subject in the control community [2], [3]. In the past three decades there are drastic theoretical advances in this study area and we could say that set invariance and Lyapunov theories play an instrumental role for such development [14].

The most popular class of invariant sets is that of the ellipsoidal ones [1]. They result from the level sets of quadratic Lyapunov functions. The reason of the popularity of the quadratic forms as invariant sets and/or candidate Lyapunov functions is clearly due to the existence of the LMI technique. The combination of quadratic functions and LMIs provides a powerful tool, that can be used to reformulate several control problems as convex semi-definite programs (SDPs). However, while the LMI technique is well appreciated, quadratic functions are quite conservative in terms of both Domain of Attraction (DoA) and robustness margin. The invariant ellipsoids, that are used to approximate the DoA, can be relatively small. In addition, it is well known [21] that there are stable systems which are not quadratically stable and stabilizable systems which are not quadratically stabilizable. Hence requiring the existence of a quadratic function can be restrictive.

To reduce the conservativeness, many non-quadratic Lyapunov functions have been considered in the literature. For stability analysis of discrete-time systems with time-varying parametric uncertainties, parameter-dependent Lyapunov functions were proposed in [7]. The conditions are expressed as LMI constraints. However, since the Lyapunov functions depend on the uncertain parameters, it is not clear how to estimate the DoA with this framework. Homogeneous Lyapunov function is another well-known solution to perform robust stability analysis and to estimate the DoA [6]. However, to the best of the author's knowledge, this approach is applicable only for continuous-time systems.

Among the classes of non-quadratic functions, the polyhedral ones are in particular interesting. Using polyhedral sets, necessary and sufficient conditions were proposed for robust stability [5]. They can also be used for robust stabilization via, e.g., vertex control [3], [10], [19]. However, it is well known [3] that constructing a polyhedral invariant set is not a trivial task, especially for high dimensional systems.

Another interesting class of non-quadratic functions is the convex hull of ellipsoids [12], [16], [20]. Using this approach, it was shown that the estimated DoA can be significantly larger than that is obtained with the quadratic Lyapunov function. However in [12], [16], [20], the associated ellipsoids are required to be robustly invariant. This implies that the considered systems are assumed to be quadratically stable/stabilizable. In [8], the quadratic stability assumption is removed. However the conditions are only for robust invariance, and are in terms of non-convex BMI constraints. In [11], the robust controlled invariance and state feedback control design problems are addressed for *continuous-time linear systems*. However, the conditions are still in terms of BMIs. In addition, it is not clear how to implement the control law in [11].

In this paper we follow the research line of [8], [11]. The aim is to provide new conditions for robust invariance and robust controlled invariance of the convex hull of ellipsoids. We address also the problem of designing a nonlinear state feedback control law. The contributions are:

1) The associated ellipsoids are not required to be robustly invariant. Hence the system is not assumed to be robustly quadratically stable/stabilizable.
2) The conditions are expressed as convex LMI constraints. Recall that the conditions in [8], [11] are in terms of non-convex BMIs.
3) The conditions can be loselessly extended to stabilization, providing a convex LMI solution to the nonlinear state feedback design problem.
4) The implementation of the new control law requires the solution of a convex optimization problem at each time instant. A tailored efficient solver is proposed to deal with the new optimization problem.

In a companion paper, it is shown that the control law is a continuous piecewise linear function of state.

The paper is organized as follows. Section II is dedicated to the robustly invariant set computation. Then in Section III results on the design of the new stabilizing nonlinear control law are presented. Section IV is concerned with the online optimization problem. Two simulated examples with comparison to earlier solutions from the literature are evaluated in Section V before drawing the conclusions in Section VI.

† SAMOVAR, Télécom-SudParis, Institut Polytechnique de Paris, 91120 Palaiseau, France hoai-nam.nguyen@telecom-sudparis.eu

Notation: A positive definite matrix P is denoted by $P \succ 0$. We denote by $\mathbb{R}^{n\times m}$ the set of real $n \times m$ matrices, and by $\mathbb{S}^n$ the set of positive definite $n \times n$ matrices. For symmetric matrices, the symbol $(*)$ denotes each of its symmetric block. For a given $P \in \mathbb{S}^n$, $\mathcal{E}(P)$ represents the following ellipsoid

$$\mathcal{E}(P) = \{x \in \mathbb{R}^n : x^T P^{-1} x \leq 1\} \tag{1}$$

We denote by $\mathbf{0}_n/\mathbf{I}_n$ the $n \times n$ zero/identity matrices. The convex hull of the sets $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_s$ is denoted as

$$\mathcal{C} = \text{Co}_{i=1}^{s} \{\mathcal{C}_i\} \tag{2}$$

For any $x \in \mathcal{C}$, $\exists v_i \in \mathcal{C}_i, \lambda_i \geq 0 : \sum_{i=1}^{s} \lambda_i = 1$ and $x = \sum_{i=1}^{s} \lambda_i v_i$. For a given set $\mathcal{C}$, its frontier is denoted as $\text{Fr}(\mathcal{C})$.

II. Robustly Invariant Set

This section studies LMI conditions for robust invariance of the convex hull of ellipsoids. This case is addressed first to introduce the main ideas. Consider the autonomous system

$$x(k+1) = A(k)x(k) \tag{3}$$

where $x(k) \in \mathbb{R}^{n_x}$ is the measured state. $A(k)$ satisfy

$$A(k) = \sum_{i=1}^{s} \alpha_i(k) A_i, \ \sum_{i=1}^{s} \alpha_i(k) = 1, \ \alpha_i(k) \geq 0 \tag{4}$$

where $A_i \in \mathbb{R}^{n_x \times n_x}, \forall i = \overline{1,s}$ are known matrices. $\alpha(k) = [\alpha_1(k) \ \ldots \ \alpha_s(k)]^T$ is a vector of uncertain and time-varying parameters.

The state $x(k)$ is subject to polytopic constraints

$$x \in \mathcal{X}, \mathcal{X} = \{x \in \mathbb{R}^{n_x} : F_x x \leq g_x\} \tag{5}$$

where $F_x \in \mathbb{R}^{c_x \times n_x}, g_x \in \mathbb{R}^{c_x}$, $g_x > 0$. The inequalities are taken component-wise.

Definition 1: (Robustly Invariant Constraint-Admissible Set) A set $\mathcal{C} \subseteq \mathbb{R}^{n_x}$ is robustly invariant for (3) if and only if $\forall x(k) \in \mathcal{C}$, it holds that $x(k+1) \in \mathcal{C}, \forall k \geq 0$. In addition, if $\mathcal{C} \subseteq \mathcal{X}$, then $\mathcal{C}$ is constraint-admissible with respect to (5).

The existence of a robustly invariant constraint-admissible set $\mathcal{C}$ for (3), (5) implies that (3) is robustly stable. In addition, if there is no initial condition $x(0) \in \text{Fr}(\mathcal{C})$ such that $x(k) \in \text{Fr}(\mathcal{C}), \forall k \geq 1$, then (3) is robustly asymptotically stable.

The following theorem holds

Theorem 1: If there exist matrices $P_i \in \mathbb{S}^{n_x}, Z_i \in \mathbb{S}^{c_x}$, $i = \overline{1,s}$, that satisfy the following LMIs

$$\begin{bmatrix} P_i & A_i P_j \\ (*) & P_j \end{bmatrix} \succeq 0, \forall i,j = \overline{1,s} \tag{6}$$

$$\begin{bmatrix} Z_i & F_x P_i \\ (*) & P_i \end{bmatrix} \succeq 0, \forall i = \overline{1,s} \tag{7}$$

$$Z_i(l) \leq g_x(l)^2, \forall l = \overline{1,c_x}, \tag{8}$$

where $Z_i(l)$ is the lth diagonal element of Z_i, then $\text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$, $i = \overline{1,s}$ is robustly invariant for system (3), and constraint-admissible with respect to (5).

Proof: See Appendix A. □

Remark 1: Using the proof of Theorem 1, it follows that conditions (6) are used to guarantee the robust invariance of $\text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$, while conditions (7), (8) are for the constraint admissibility of $\text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$. □

Remark 2: If $P_i = P$, $\forall i = \overline{1,s}$, then conditions (6) are the robust invariance conditions of the single ellipsoid $\mathcal{E}(P)$. Hence quadratic invariance is a special case of (6). □

Remark 3: The number of LMIs in (6), (7), (8) increases quadratically as s increases. However, the number of LMIs is not affected by the system dimension, that can be high dimensional. □

Theorem 1 provides LMI conditions for robust invariance and constraint admissibility of $\text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$. Among all $\text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$, we would like to obtain the largest one. This can be done by maximizing the volume of the associated ellipsoids. Another idea is to optimize the ellipsoids with respect to some reference directions or sets [13]. Here we show how to optimize the set with respect to a reference direction.

Let d_i be a reference direction for $\mathcal{E}(P_i)$, $\forall i = \overline{1,s}$. The problem of optimizing $\mathcal{E}(P_i)$ with respect to d_i is equivalent to maximize μ_i with the constraint $\mu_i^2 d_i^T P_i^{-1} d_i \leq 1$. Thus, using the Schur complement, one gets

$$\begin{bmatrix} 1 & \mu_i d_i^T \\ (*) & P_i \end{bmatrix} \succeq 0, \forall i = \overline{1,s} \tag{9}$$

Combining (6), (7), (8), (9), one can write the problem of maximizing the size of $\text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$, $\forall i = \overline{1,s}$ by

$$\begin{aligned} &\max_{P_i, Z_i, \mu_i} \left\{ \sum_{i=1}^{s} \mu_i \right\}, \\ &\text{s.t. } (6), (7), (8), (9) \end{aligned} \tag{10}$$

Problem (10) is a convex SDP program. It can be solved efficiently using free available LMI parser such as CVX [9] or Yalmip [15].

III. Robustly Controlled Invariant Set and Nonlinear State Feedback Control Design

A. Robustly Controlled Invariant Set

Perhaps the most interesting aspect of the conditions in Theorems 1 is that they can be loselessly extended to compute a robustly controlled invariant set, and to design a nonlinear state feedback control law. To this aim, let us consider the following system

$$x(k+1) = A(k)x(k) + B(k)u(k) \tag{11}$$

where $u(k) \in \mathbb{R}^{n_u}$ is the control input. $A(k)$ is defined in (4), $B(k) = \sum_{i=1}^{s} \alpha_i(k) B_i$, where $B_i \in \mathbb{R}^{n_x \times n_u}$ are known, $\alpha(k)$ are defined in (4). The constraints on $x(k)$ are given in (5). The constraints on $u(k)$ are

$$u \in \mathcal{U} = \{u \in \mathbb{R}^{n_u} : F_u u \leq g_u\} \tag{12}$$

where $F_u \in \mathbb{R}^{c_u \times n_u}$ and $g_u \in \mathbb{R}^{c_u}$ are constant with $g_u > 0$.

Definition 2: (Robustly Controlled Invariant Constraint-Admissible Set) A set $\mathcal{C} \subseteq \mathbb{R}^{n_x}$ is robustly controlled invariant for (11) if and only if $\forall x(k) \in \mathcal{C}$, $\exists u(k) = u(x(k))$, one has $x(k+1) \in \mathcal{C}, \forall k \geq 0$. In addition, if $\mathcal{C} \subseteq \mathcal{X}$ and $u(x(k)) \in \mathcal{U}$, then $\mathcal{C}$ is constraint-admissible with respect to (5), (12).

The following theorem holds

Theorem 2: If there exist matrices $P_i \in \mathbb{S}^{n_x}, Z_i \in \mathbb{S}^{c_x}$, $G_i \in \mathbb{S}^{c_u}$, $Y_i \in \mathbb{R}^{n_u \times n_x}, \forall i = \overline{1,s}$ such that the following LMIs

$$\begin{bmatrix} P_i & (A_iP_j + B_iY_j) \\ (*) & P_j \end{bmatrix} \succeq 0, \forall i,j = \overline{1,s} \quad (13)$$

$$\begin{bmatrix} Z_i & F_xP_i \\ (*) & P_i \end{bmatrix} \succeq 0, \forall i = \overline{1,s} \quad (14)$$

$$Z_i(l) \leq g_x(l)^2, \forall l = \overline{1,c_x} \quad (15)$$

$$\begin{bmatrix} G_i & F_uY_i \\ (*) & P_i \end{bmatrix} \succeq 0, \forall i = \overline{1,s} \quad (16)$$

$$G_i(l) \leq g_u(l)^2, \forall l = \overline{1,c_u} \quad (17)$$

are satisfied, then $\text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$ is robustly controlled invariant constraint-admissible for (11), (5), (12).

Proof: See Appendix B. □

Remark 4: Using the proof of Theorem 2, it follows that $x(k)$ is keeping inside $\text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$ with the control law

$$u(k) = \sum_{i=1}^{s} \lambda_i(k) K_i v_i(k) \quad (18)$$

where $K_i = Y_iP_i^{-1}$, $v_i(k)$ and $\lambda_i(k)$ satisfy

$$\sum_{i=1}^{s} \lambda_i(k) v_i(k) = x(k) \quad (19)$$

with $v_i(k) \in \mathcal{E}(P_i)$, $\lambda_i \geq 0$, $\sum_{i=0}^{s} \lambda_i(k) = 1$. Note that $\lambda_i(k)$, $\forall i = \overline{1,s}$ are not the uncertain parameters $\alpha(k)$ in (4). They are the interpolating coefficients, that are used to decompose $x(k)$ into several *auxiliary* states $v_i(k)$. □

Remark 5: Consider the case $K_i = K, \forall i = \overline{1,s}$. Using (18), one has

$$u(k) \quad = \sum_{i=1}^{s} \lambda_i(k) K_i v_i(k) = K \left(\sum_{i=1}^{s} \lambda_i(k) v_i(k) \right)$$

Thus, using (19), one obtains $u(k) = Kx(k)$. Hence, linear feedback control law is a particular case of (18), (19). □

Similarly to Section II, once robust controlled invariance and constraint admissibility conditions are expressed as LMI constraints (13), (14), (15), (16), (17), the set $\text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$ can be optimized by solving the following SDP problem

$$\max_{P_i,Z_i,G_i,Y_i,\mu_i} \left\{ \sum_{i=1}^{s} \mu_i \right\}, \quad \text{s.t. } (13),(14),(15),(16),(17),(9) \quad (20)$$

B. Nonlinear State Feedback Control Design

In this section, it is assumed that the LMIs (13) are strict. This implies that (11) is robustly asymptotically stabilizable. As shown with the proof of Theorem 2, recursive feasibility is guaranteed with the control law

$$u(k) = \sum_{i=1}^{s} \lambda_i(k) K_i v_i(k) \quad (21)$$

where $v_i(k) \in \mathcal{E}(P_i)$, $\lambda_i(k) \geq 0$, $\sum_{i=1}^{s} \lambda_i(k) = 1$ and

$$\sum_{i=1}^{s} \lambda_i(k) v_i(k) = x(k) \quad (22)$$

Here by recursive feasibility, we mean that $x(k+1) \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i)), \forall x(k) \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$.

For a given $x(k) \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$, there are generally several $v_i(k), \lambda_i(k)$ satisfying (22), see Fig. 1. The variables $v_i(k), \lambda_i(k)$ can be considered as the degrees of freedom. Here we show how to select $v_i(k), \lambda_i(k)$ in such a way that the control law (21) robustly asymptotically stabilizes (11).

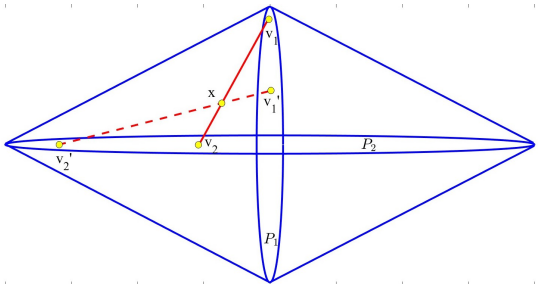


Fig. 1: State Decomposition.

For a given x, consider the following optimization problem

$$\min_{v_i,\lambda_i} \{ \sum_{i=1}^{s} \lambda_i \}, \quad \text{s.t.} \begin{cases} \sum_{i=1}^{s} \lambda_i v_i = x, \\ v_i^T P_i^{-1} v_i \leq 1, \forall i = \overline{1,s}, \\ \lambda_i \geq 0, \forall i = \overline{1,s} \end{cases} \quad (23)$$

Note that we do not consider the constraint $\sum_{i=1}^{s} \lambda_i = 1$ in (23). Clearly, $\min\{\sum_{i=1}^{s} \lambda_i\} \leq 1$, $\forall x \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$. Note also that (23) is a non-convex optimization problem due to the multiplications between λ_i and v_i. We will provide a way to solve (23) efficiently in the next section. For the moment, let us assume that the solution of (23) is available, and is denoted as λ_i^*, v_i^*.

We have the following theorem.

Theorem 3: The control law (21) with $\lambda_i(k) = \lambda_i^*$ and $v_i(k) = v_i^*$ guarantees recursive feasibility and robust asymptotic stability $\forall x(k) \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$.

Proof: See Appendix C. □

In summary, the proposed control strategy consists of two stages: offline stage and online stage.

Algorithm 1: Offline Stage

1: Calculate P_i, $Y_i, \forall i = \overline{1,s}$ by solving (20).
2: Calculate the gains K_i as, $K_i = Y_iP_i^{-1}$, $\forall i = \overline{1,s}$.

Algorithm 1: Online Stage - At Time Instant k

1: Measure or estimate $x(k)$.
2: Solve the optimization problem (23) to obtain λ_i^*, v_i^*.
3: The control law is given as $u(k) = \sum_{i=1}^{s} \lambda_i^* K_i v_i^*$.

IV. Online Optimization Problem

A. SDP Formulation

The implementation of the control law (21), (23) requires the solution of a non-convex problem at each time instant. In the following, we show how to reformulate (23) as a convex SDP problem. For this purpose, define

$$\zeta_i = \lambda_i v_i, \forall i = \overline{1,s} \tag{24}$$

Using (24), problem (23) becomes

$$\begin{array}{l} \min\limits_{\zeta_i,\lambda_i}\{\sum\limits_{i=1}^{s}\lambda_i\}, \\ \text{s.t.} \left\{ \begin{array}{l} \sum\limits_{i=1}^{s}\zeta_i = x, \\ \zeta_i^T P_i^{-1}\zeta_i \le \lambda_i^2, \forall i = \overline{1,s}, \\ \lambda_i \ge 0, \forall i = \overline{1,s} \end{array} \right. \end{array} \tag{25}$$

Thus, using the Schur complement

$$\begin{array}{l} \min\limits_{\zeta_i,\lambda_i}\{\sum\limits_{i=1}^{s}\lambda_i\}, \\ \text{s.t.} \left\{ \begin{array}{l} \sum\limits_{i=1}^{s}\zeta_i = x, \\ \left[\begin{array}{cc} \lambda_i & \zeta_i^T \\ \zeta_i & \lambda_i P_i \end{array} \right] \succeq 0, \forall i = \overline{1,s}, \\ \lambda_i \ge 0, \forall i = \overline{1,s} \end{array} \right. \end{array} \tag{26}$$

(26) is a convex SDP problem, which is required to solve at each time instant. Although there has been a lot of progress in convex optimization, solving a SDP problem in an embedded platform can generally take an unnecessary long time. This might restrict the applicability of the approach for fast dynamical systems. In this next section, we provide a way to solve efficiently (26), or equivalently (25). The solution is based on the alternating direction method of multipliers (ADMM) algorithm. In the past ten years, ADMM has emerged as a powerful algorithm for solving structured convex optimization problems. The main contribution of this section is to show how to convert (25) into a form that the sub-optimization problems associated with the ADMM can be solved efficiently.

In the following, the ADMM theory is recalled [4].

B. Alternating Direction Method of Multipliers

Consider the optimization problem

$$\begin{array}{l} \min\limits_{z,w}\left\{f(z)+h(w)\right\}, \\ \text{s.t. } z - w = \mathbf{0} \end{array} \tag{27}$$

where $f(z), h(w)$ are convex functions. One way to solve (27) is to form the following augmented Lagrangian

$$\mathcal{L}_\rho(z,w,\theta) = f(z)+h(w)+\theta^T(z-w)+\frac{\rho}{2}(z-w)^T\mathcal{P}(z-w) \tag{28}$$

where θ is the Lagrange multiplier, $\rho \ge 0$ is a tuning parameter, and $\mathcal{P}$ is a positive definite matrix. In general, $\mathcal{P}$ is chosen as $\mathcal{P} = \mathbf{I}$ [4]. However, $\mathcal{P}$ is a design parameter in this paper. ρ controls the relative priority between the cost function and the constraints. In practice, trial and error can be used to select a good value for ρ.

In each iteration of ADMM, we perform alternating minimization of $\mathcal{L}_\rho(z,w,\theta)$ over z and w. At iteration q we carry out the following steps.

$$z^{(q+1)} := \arg\min_z\{\mathcal{L}_\rho(z, w^{(q)}, \theta^{(q)})\} \tag{29}$$

$$w^{(q+1)} := \arg\min_w\{\mathcal{L}_\rho(z^{(q+1)}, w, \theta^{(q)})\} \tag{30}$$

$$\theta^{(q+1)} := \theta^{(q)} + \rho\mathcal{P}(z^{(q+1)} - w^{(q+1)}) \tag{31}$$

where the superscript (q) is used to denote the values of variables calculated at iteration q.

The ADMM is particularly useful when the minimizations (29), (30) can be carried out efficiently. In the next section we show how to convert (25) into a form that (29), (30) admit closed-form expressions.

The primal and dual residuals at iteration q are given by

$$e_p^{(q)} = z^{(q)} - w^{(q)},\ e_d^{(q)} = z^{(q)} - z^{(q-1)} \tag{32}$$

The algorithm is terminated when the primal and dual residuals satisfy a stopping criterion. A typical criterion is to stop when $\|e_p^{(q)}\|_\infty \le \epsilon_p$, $\|e_d^{(q)}\|_\infty \le \epsilon_d$, where ϵ_d, ϵ_p are given tolerances.

C. Efficient ADMM-based Solver

In this section, we will show how cast (25) as an optimization problem (27). The following definition is recalled [4].

Definition 3: (Indicator function) For a given set $\mathcal{C} \subset \mathbb{R}^{n_x}$, its indicator function is denoted by $\mathcal{I}_\mathcal{C}(x) : \mathbb{R}^{n_x} \to \{0, \infty\}$ where

$$\mathcal{I}_\mathcal{C}(x) = \left\{ \begin{array}{ll} 0 & \text{if } x \in \mathcal{C}, \\ +\infty & \text{if } x \notin \mathcal{C} \end{array} \right.$$

Let us take

$$z = [\lambda_1\ \lambda_2\ \dots\ \lambda_s\ \zeta_1^T\ \zeta_2^T\ \dots\ \zeta_s^T]^T \tag{33}$$

Define $\underline{\lambda}_i, \underline{\zeta}_i$, and w as

$$\left\{ \begin{array}{l} \underline{\lambda}_i = \lambda_i,\ \underline{\zeta}_i = \zeta_i,\ \forall i = \overline{1,s} \\ w = [\underline{\lambda}_1\ \underline{\lambda}_2\ \dots\ \underline{\lambda}_s\ \underline{\zeta}_1^T\ \underline{\zeta}_2^T\ \dots\ \underline{\zeta}_s^T]^T \end{array} \right. \tag{34}$$

The matrix $\mathcal{P}$ is chosen as

$$\mathcal{P} = \text{diag}\left(1, \dots, 1, P_1^{-1}, \dots, P_s^{-1}\right) \tag{35}$$

Using (33), (34), problem (25) is recast as (27) with

$$\left\{ \begin{array}{l} f(z) = \sum\limits_{i=1}^{s}\lambda_i + \mathcal{I}_{\sum\limits_{i=1}^{s}\zeta_i = x}(\lambda, \zeta), \\ h(w) = \sum\limits_{i=1}^{s}\mathcal{I}\left\{ \begin{array}{l} \underline{\zeta}_i^T P_i^{-1}\underline{\zeta}_i \le \underline{\lambda}_i^2, \\ \underline{\lambda}_i \ge 0 \end{array} \right. (\underline{\lambda}_i, \underline{\zeta}_i) \end{array} \right. \tag{36}$$

Applying ADMM to problem (27), (36), we carry out the following steps in each iteration.

Step 1: The optimization problem is

$$\begin{array}{l} \min\limits_z\left\{\frac{\rho}{2}z^T\mathcal{P}z + f_z^T z\right\} \\ \text{s.t. } C_z z = x \end{array} \tag{37}$$

where

$$\begin{array}{l} f_z = f_\lambda + \theta^{(q)} - \rho\mathcal{P}w^{(q)}, \\ f_\lambda = [1\ 1\ \dots\ 1\ \mathbf{0}\ \mathbf{0}\ \dots\ \mathbf{0}]^T, \\ C_z = \left[\begin{array}{cccccccc} \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{I} & \mathbf{I} & \dots & \mathbf{I} \end{array} \right] \end{array} \tag{38}$$

(37) is a quadratic program with equality constraints. Using the method of Lagrange multipliers, the optimal solution of (37) is given as [4]

$$\begin{aligned} z^{(q+1)} &= S_x x + S_f f_z \\ S_x &= \mathcal{P}^{-1} C_z^T (C_z \mathcal{P}^{-1} C_z^T)^{-1}, \\ S_f &= \rho^{-1} \mathcal{P}^{-1} (-\mathbf{I} + C_z^T (C_z \mathcal{P}^{-1} C_z^T)^{-1} C_z \mathcal{P}^{-1}) \end{aligned} \tag{39}$$

The matrices S_f, S_x are constant, and are calculated offline.

Step 2: The optimization problem is

$$\begin{aligned} &\min_{w} \left\{ \tfrac{\rho}{2} w^T \mathcal{P} w - (\theta^{(q)} + \rho \mathcal{P} z^{(q+1)})^T w \right\}, \\ &\text{s.t.} \begin{cases} \underline{\zeta}_i^T P_i^{-1} \underline{\zeta}_i \le \underline{\lambda}_i^2, \\ \underline{\lambda}_i \ge 0 \end{cases} \end{aligned} \tag{40}$$

The cost function and the constraints of (40) are separable in $(\underline{\lambda}_i, \underline{\zeta}_i)$, $\forall i = \overline{1, s}$. Their update can all be carried out in parallel. The solution of the following problem provides the update of $(\underline{\lambda}_i, \underline{\zeta}_i)$

$$\begin{aligned} &\min_{\underline{\lambda}_i, \underline{\zeta}_i} \Big\{ \tfrac{\rho}{2} \underline{\lambda}_i^2 + \tfrac{\rho}{2} \underline{\zeta}_i^T P_i^{-1} \underline{\zeta}_i - (\theta_{\underline{\lambda}_i}^{(q)} + \rho \lambda_i^{(q+1)}) \underline{\lambda}_i \\ &\qquad - (\theta_{\underline{\zeta}_i}^{(q)} + \rho P_i^{-1} \zeta_i^{(q+1)})^T \underline{\zeta}_i \Big\}, \\ &\text{s.t.} \begin{cases} \underline{\zeta}_i^T P_i^{-1} \underline{\zeta}_i \le \underline{\lambda}_i^2, \\ \underline{\lambda}_i \ge 0 \end{cases} \end{aligned} \tag{41}$$

where $\theta_{\underline{\lambda}_i}^{(q)}, \theta_{\underline{\zeta}_i}^{(q)}$ are, respectively, the Langrange multipliers for $\underline{\lambda}_i, \underline{\zeta}_i$ at iteration q. Rewrite problem (41) as

$$\begin{aligned} &\min_{\underline{\lambda}_i, \underline{\zeta}_i} \left\{ (\underline{\lambda}_i - \beta_i)^2 + (\underline{\zeta}_i - \gamma_i)^T P_i^{-1} (\underline{\zeta}_i - \gamma_i) \right\}, \\ &\text{s.t.} \begin{cases} \underline{\zeta}_i^T P_i^{-1} \underline{\zeta}_i \le \underline{\lambda}_i^2, \\ \underline{\lambda}_i \ge 0 \end{cases} \end{aligned} \tag{42}$$

where $\beta_i = \frac{1}{\rho} \theta_{\underline{\lambda}_i}^{(q)} + \lambda_i^{(q+1)}$, $\gamma_i = \frac{P_i}{\rho} \theta_{\underline{\zeta}_i}^{(q)} + \zeta_i^{(q+1)}$ The solution of (42) can be computed explicitly by using the method of Lagrange multipliers. The solution is given by [18]

$$\begin{cases} \text{If } \beta_i < 0, \gamma_i^T P_i^{-1} \gamma_i \le \beta_i^2, \\ \quad \underline{\lambda}_i^{(q+1)} = 0, \underline{\zeta}_i^{(q+1)} = \mathbf{0} \\ \text{If } \beta_i \ge 0, \gamma_i^T P_i^{-1} \gamma_i \le \beta_i^2, \\ \quad \underline{\lambda}_i^{(q+1)} = \beta_i, \underline{\zeta}_i^{(q+1)} = \gamma_i \\ \text{Otherwise,} \\ \quad \underline{\lambda}_i^{(q+1)} = \frac{\beta_i + \sqrt{\gamma_i^T P_i^{-1} \gamma_i}}{2}, \\ \quad \underline{\zeta}_i^{(q+1)} = \frac{\beta_i + \sqrt{\gamma_i^T P_i^{-1} \gamma_i}}{2\sqrt{\gamma_i^T P_i^{-1} \gamma_i}} \gamma_i \end{cases} \tag{43}$$

Remark 6: The solution of (42) can be obtained analytically by using $\mathcal{P}$ in (35). If $\mathcal{P} = \mathbf{I}$ is used as in the *standard* ADMM [4], then the solution of (42) can only be obtained numerically via iterative procedures. □

Algorithm 2 summarizes the particularization of the ADMM method to problem (25).

Algorithm 2: ADMM based Solver

1: Using (39) to update $z^{(q+1)}$.
2: Using (43) to update $w^{(q+1)}$.
3: $\theta^{(q+1)} \leftarrow \theta^{(q)} + \rho \mathcal{P}(z^{(q+1)} - w^{(q+1)})$.

V. Example

This section demonstrate the potential benefit of the new methods by simulations of two examples system. The CVX toolbox [9] was used to solve SDP problems.

A. Example 1

Consider system (3) with

$$A_1 = \begin{bmatrix} 0.9022 & 0.0085 \\ -0.0036 & 0.9858 \end{bmatrix}, \; A_2 = \begin{bmatrix} 1.2 & 0.6 \\ -0.5 & 0 \end{bmatrix} \tag{44}$$

The constraints are

$$-1 \le [0.0036 \; 0.0142] x \le 1, \; -145 \le x_1 \le 145$$

The goal is to calculate a robust invariant constraint-admissible set that is as large as possible.

Using Theorem 1, Fig. 2 shows the ellipsoids $\mathcal{E}(P_1)$ (solid blue line), $\mathcal{E}(P_2)$ (solid red line). The matrices P_1, P_2 are

$$P_1 = 10^4 \begin{bmatrix} 1.9455 & -0.6858 \\ -0.6858 & 0.7173 \end{bmatrix}, \\ P_2 = 10^4 \begin{bmatrix} 2.1024 & -0.8832 \\ -0.8832 & 0.6895 \end{bmatrix}$$

For comparison, Fig. 2 also shows the maximum-volume robustly invariant constraint-admissible ellipsoid (dash-dot black line), and the maximal robustly invariant constraint-admissible polytope (dashed green line). It can be observed that $\text{Co}_{i=1}^2(\mathcal{E}(P_i))$ is slightly smaller than the polytope, and both of them are much larger than the maximum-volume ellipsoid.

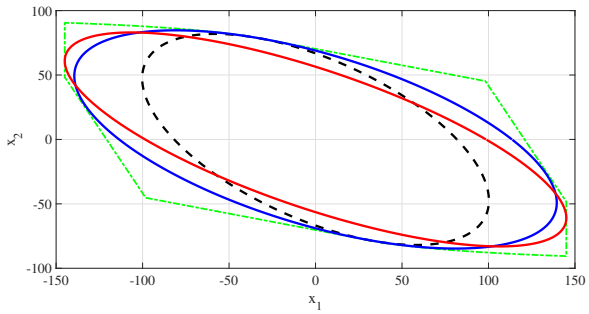


Fig. 2: Robustly invariant constraint-admissible sets. Convex hull of ellipsoids (solid blue and solid red), maximum-volume ellipsoid (dash-dot black), and maximal polyhedral set (dashed green) for example 1.

B. Example 2

Consider the system (11) with

$$A_1 = \begin{bmatrix} 1.0 & -1.4 \\ -1.0 & -0.8 \end{bmatrix} \; A_2 = \begin{bmatrix} 1.0 & 1.4 \\ -1.0 & -0.8 \end{bmatrix}, \\ B_1 = \begin{bmatrix} 5.9 & 2.8 \end{bmatrix}^T, \; B_2 = \begin{bmatrix} 3.1 & -2.8 \end{bmatrix}^T \tag{45}$$

The goal is to design a robust stabilizing controller. There are only input constraints $-1 \le u(k) \le 1$.

It can be verified that (45) is not quadratically stabilizable. Using Theorem 2, one obtains

$$P_1 = \begin{bmatrix} 22.91 & -19.99 \\ -19.99 & 18.01 \end{bmatrix}, \; P_2 = \begin{bmatrix} 21.24 & -10.51 \\ -10.51 & 8.97 \end{bmatrix},$$
$$K_1 = \begin{bmatrix} -0.0949 & 0.1296 \end{bmatrix}, \; K_2 = \begin{bmatrix} -0.0095 & 0.2060 \end{bmatrix}$$

It is interesting to observed that the closed-loop system with the linear control law $u(k) = K_1 x(k)$ or with $u(k) = K_2 x(k)$ is not robustly stable. For example, the eigenvalues of $(A_1 + B_1 K_2)$ are 1.0887 and -0.3676.

The sets $\mathcal{E}(P_1)$, $\mathcal{E}(P_2)$ are presented in Fig. 3. Note that

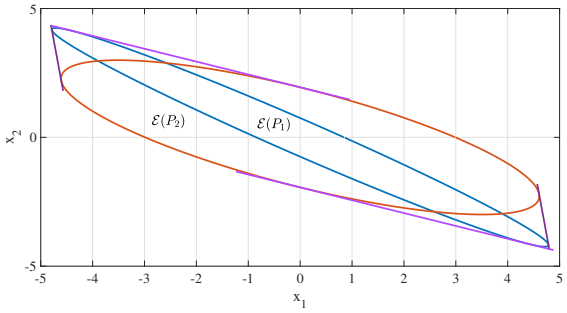
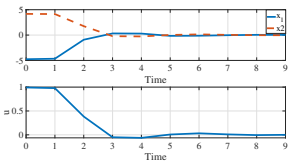


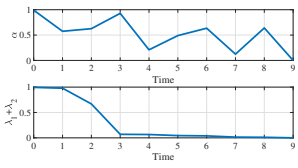
Fig. 3: Convex hull of ellipsoids for example 2.

procedures in [3] to calculate a robustly controlled invariant and constraint-admissible polyhedral set for (45) could not be terminated after one hour.

For the initial condition $x(0) = [-4.7589 \quad 4.175]^T$, Fig. 4(a) presents the state and input trajectories of the closed-loop system as functions of time. Fig. 4(b) shows the realization of α and the sum $(\lambda_1 + \lambda_2)$ as functions of time. As expected, $(\lambda_1 + \lambda_2)$ is a decreasing function.



(a) State and Input



(b) α and $\lambda_1 + \lambda_2$

Fig. 4: (a) State and input trajectories (b) α realization and $\lambda_1 + \lambda_2$ trajectories for example 2.

For this example, we solved the online optimization problem (26) using the SEDUMI solver in CVX toolbox, and using the ADMM solver in Algorithm 2. Using the TIC/TOC function of MATLAB 2022b, we found that the online computation times for one sampling interval were $0.4318[s]$, and $2.2296 \times 10^{-3}[s]$ for the SEDUMI solver, and for the ADMM solver, respectively.

VI. Conclusion

In this paper, the convex hull of ellipsoids is considered as a powerful tool for constrained control of uncertain and/or time-varying linear discrete-time systems. Using this class of set, convex LMI conditions for robust invariance and robust controlled invariance are proposed. Similar results have also established for the nonlinear state feedback control design. A tailored efficient ADMM algorithm is proposed to solve online the optimization, that is required for the implementation of the new control law. Two numerical examples demonstrate the effectiveness of the new approach.

The proposed approach is a natural generalization of the quadratic function framework in the robust case. We believe it will provide solution for several control design problems that have not been found a definitive answer. This claim is supported by the fact that using the given framework a set of control gains is designed. Each control gain in the set takes charge of a local region in the state space via its associated ellipsoid.

References

[1] Francesco Amato. *Robust control of linear systems subject to uncertain time-varying parameters*, volume 325. Springer, 2006.

[2] B Ross Barmish and EI Jury. New tools for robustness of linear systems. *IEEE Transactions on Automatic Control*, 39(12):2525–2525, 1994.

[3] Franco Blanchini and Stefano Miani. *Set-theoretic methods in control*, volume 78. Springer, 2008.

[4] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

[5] Robert Brayton and Christopher Tong. Constructive stability and asymptotic stability of dynamical systems. *IEEE Transactions on Circuits and Systems*, 27(11):1121–1130, 1980.

[6] Graziano Chesi, Andrea Garulli, Alberto Tesi, and Antonio Vicino. Homogeneous lyapunov functions for systems with structured uncertainties. *Automatica*, 39(6):1027–1035, 2003.

[7] Jamal Daafouz and Jacques Bernussou. Parameter dependent lyapunov functions for discrete time systems with time varying parametric uncertainties. *Systems & control letters*, 43(5):355–359, 2001.

[8] Rafal Goebel, Tingshu Hu, and Andrew R Teel. Dual matrix inequalities in stability and performance analysis of linear differential/difference inclusions. *Current Trends in Nonlinear Systems and Control: In Honor of Petar Kokotović and Turi Nicosia*, pages 103–122, 2006.

[9] Michael Grant, Stephen Boyd, and Yinyu Ye. Cvx: Matlab software for disciplined convex programming, 2008.

[10] P-O Gutman and Michael Cwikel. Admissible sets and feedback control for discrete-time linear dynamical systems with bounded controls and states. *IEEE transactions on Automatic Control*, 31(4):373–376, 1986.

[11] Tingshu Hu. Nonlinear control design for linear differential inclusions via convex hull of quadratics. *Automatica*, 43(4):685–692, 2007.

[12] Tingshu Hu and Zongli Lin. Composite quadratic lyapunov functions for constrained control systems. *IEEE Transactions on Automatic Control*, 48(3):440–450, 2003.

[13] Tingshu Hu, Zongli Lin, and Ben M Chen. Analysis and design for discrete-time linear systems subject to actuator saturation. *Systems & control letters*, 45(2):97–112, 2002.

[14] Hassan K Khalil. Nonlinear systems. *Patience Hall*, 115, 2002.

[15] Johan Lofberg. Yalmip: A toolbox for modeling and optimization in matlab. In *2004 IEEE international conference on robotics and automation (IEEE Cat. No. 04CH37508)*, pages 284–289. IEEE, 2004.

[16] Hoai-Nam Nguyen. Constrained control of uncertain, time-varying, discrete-time systems. *Lecture Notes in Control and Infomration Sciences*, 451:17, 2014.

[17] Hoai-Nam Nguyen. Optimizing prediction dynamics with saturated inputs for robust model predictive control. *IEEE Transactions on Automatic Control*, 66(1):383–390, 2020.

[18] Hoai-Nam Nguyen. Improved prediction dynamics for robust mpc. *IEEE Transactions on Automatic Control*, 2022.

[19] Hoai-Nam Nguyen, Per-Olof Gutman, Sorin Olaru, and Morten Hovd. Implicit improved vertex control for uncertain, time-varying linear discrete-time systems with state and control constraints. *Automatica*, 49(9):2754–2759, 2013.

[20] Hoai-Nam Nguyen, Sorin Olaru, Per Olof Gutman, and Morten Hovd. Constrained control of uncertain, time-varying linear discrete-time systems subject to bounded disturbances. *IEEE Transactions on Automatic Control*, 60(3):831–836, 2014.

[21] A Olas. On robustness of systems with structured uncertainties. In *Mechanics and Control*, pages 156–169. Springer, 1992.

APPENDIX

A. Proof of Theorem 1: For robust invariance proof, one needs to show that $\forall x(k) \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$, one has $x(k+1) \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$, $\forall k \geq 0$. We decompose $x(k)$ as

$$x(k) = \sum_{j=1}^{s} \lambda_j(k) v_j(k) \tag{46}$$

where $v_j(k) \in \mathcal{E}(P_j)$, $\lambda_j(k) \geq 0$ and $\sum_{j=1}^{s} \lambda_j(k) = 1$.

Using (46), one has

$$\begin{aligned} x(k+1) &= A(k)x(k) = A(k)\sum_{j=1}^{s} \lambda_j(k) v_j(k) \\ &= \sum_{j=1}^{s} \lambda_j(k) A(k) v_j(k) \end{aligned}$$

or equivalently, $v_j(k+1) = A(k)v_j(k)$

$$x(k+1) = \sum_{j=1}^{s} \lambda_j(k) v_j(k+1) \tag{47}$$

Using (47), if we can show that

$$v_j(k+1) \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i)), j = \overline{1,s} \tag{48}$$

then $x(k+1) \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$. Pre and post multiplying (6) by

$$\begin{bmatrix} \mathbf{I}_{n_x} & \mathbf{0}_{n_x} \\ \mathbf{0}_{n_x} & P_j^{-1} \end{bmatrix}$$

one obtains, $\forall j = \overline{1,s}$

$$\begin{bmatrix} P_i & A_i \\ A_i^T & P_j^{-1} \end{bmatrix} \succeq 0, \forall i,j = \overline{1,s}$$

For each i, multiply the corresponding by $\alpha_i(k)$, and sum to obtain, $\forall i = \overline{1,s}$

$$\begin{bmatrix} \sum_{i=1}^{s} \alpha_i(k) P_i & \sum_{i=1}^{s} \alpha_i(k) A_i \\ (*) & P_j^{-1} \end{bmatrix} \succeq 0$$

Thus, using the Schur complement

$$P_j^{-1} \succeq A(k)^T \left(\sum_{i=1}^{s} \alpha_i(k) P_i \right)^{-1} A(k)$$

with $A(k) = \sum_{i=1}^{s} \alpha_i(k) A_i$. Or equivalently

$$\begin{aligned} &v_j(k)^T P_j^{-1} v_j(k) \geq \\ &\quad v_j(k+1)^T \left(\sum_{i=1}^{s} \alpha_i(k) P_i \right)^{-1} v_j(k+1) \end{aligned} \tag{49}$$

Using (49), it follows that $\forall v_j(k) \in \mathcal{E}(P_j)$, one has

$$v_j(k+1) \in \mathcal{E}\left(\sum_{i_l=1}^{s} \alpha_i(k) P_i \right) \tag{50}$$

where $\mathcal{E}\left(\sum_{i_l=1}^{s} \alpha_i(k) P_i \right)$ is a parameterized ellipsoid, that depends on $\alpha_i(k)$. Now we show by contradiction that

$$\mathcal{E}\left(\sum_{i_l=1}^{s} \alpha_i(k) P_i \right) \subseteq \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$$

Assume $\exists x_c \in \mathcal{E}\left(\sum_{i_l=1}^{s} \alpha_i(k) P_i \right)$ such that $x_c \notin \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$. Without loss of generality, one can assume $x_c \in \text{Fr}\left(\mathcal{E}\left(\sum_{i_l=1}^{s} \alpha_i(k) P_i \right) \right)$. Let $f_c \in \mathbb{R}^{n_x}$ be a supporting hyperplane of $\mathcal{E}\left(\sum_{i_l=1}^{s} \alpha_i(k) P_i \right)$ at x_c. Because the sets $\mathcal{E}\left(\sum_{i_l=1}^{s} \alpha_i(k) P_i \right)$ and $\text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$ are symmetric, one has

$$|f_c^T x| < |f_c^T x_c|, \forall x \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i)) \tag{51}$$

Denote $g_c = |f_c^T x_c|$. Hence $\exists \alpha_i(k) \geq 0, \sum_{i=1}^{s} \alpha_i(k) = 1$ such that

$$f_c^T \left(\sum_{i=1}^{s} \alpha_i(k) P_i \right) f_c = g_c^2 \tag{52}$$

Using (51), one gets $|f_c^T x| < g_c^2, \forall x \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$. It is well known [17] that this condition is satisfied if and only if $f_c^T P_i f_c < g_c^2$. Hence, $\forall \alpha_i(k) \geq 0, \sum_{i=1}^{s} \alpha_i(k) = 1$

$$f_c^T \left(\sum_{i=1}^{s} \alpha_i(k) P_i \right) f_c < g_c^2 \tag{53}$$

which contradicts (52). Hence $\mathcal{E}\left(\sum_{i_l=1}^{s} \alpha_i(k) P_i \right) \subseteq \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$. Using (50), one gets $v_j(k+1) \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i)), \forall v_j(k) \in \mathcal{E}(P_j), \forall j = \overline{1,s}$. It follows that the set $\text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$ is robustly invariant.

It remains to prove the constraint admissibility. Concerning the constraints (5), clearly, $\text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$ is constraint-admissible if and only if $\mathcal{E}(P_i) \subseteq \mathcal{X}$, $i = \overline{1,s}$. Using [17], $\mathcal{E}(P_i) \subseteq \mathcal{X}$ if and only if $\exists Z_i \in \mathbb{S}^{c_x}$ such that the LMIs (7), (8) are satisfied. The proof is complete. □

B. Proof of Theorem 2: For robust controlled invariance, one needs to show that $\forall x(k) \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$, $\exists u(k)$ such that $x(k+1) \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$.

Decompose $x(k) \in \text{Co}_{i=1}^{s}(\mathcal{E}(P_i))$ as

$$x(k) = \sum_{j=1}^{s} \lambda_j(k) v_j(k) \tag{54}$$

where $v_j(k) \in \mathcal{E}(P_j)$ and $\sum_{j=1}^{s} \lambda_j(k) = 1$, $\lambda_j(k) \geq 0$. Consider the following control law

$$u(k) = \sum_{j=1}^{s} \lambda_j(k) u_j(k) \tag{55}$$

where, $K_j = Y_j P_j^{-1}$

$$u_j(k) = K_j v_j(k), \forall j = \overline{1,s} \tag{56}$$

Substituting (54), (55), (56) into (11), one gets

$$\begin{aligned} & x(k+1) \\ & = \sum_{j=1}^{s} \lambda_j(k) A(k) v_j(k) + \sum_{j=1}^{s} \lambda_j(k) K_j v_j(k) \\ & = \sum_{j=1}^{s} \lambda_j(k) (A(k) + B(k) K_j) v_j(k) = \sum_{j=1}^{s} \lambda_j(k) v_j(k+1) \end{aligned} \tag{57}$$

with

$$v_j(k+1) = (A(k) + B(k) K_j)\, v_j(k) \tag{58}$$

Now we will use (13) to show that, if $v_j(k) \in \mathcal{E}(P_j)$ then $v_j(k+1) \in \mathrm{Co}_{i=1}^{s} (\mathcal{E}(P_i))$, $\forall j = \overline{1,s}$.

Rewrite (13) as, with $Y_j = K_j P_j$,

$$\begin{bmatrix} P_i & (A_i + B_i K_j)\, P_j \\ (*) & P_j \end{bmatrix} \succeq 0 \tag{59}$$

Pre and post multiplying (59) by

$$\begin{bmatrix} \mathbf{I}_{n_x} & \mathbf{0}_{n_x} \\ \mathbf{0}_{n_x} & P_j^{-1} \end{bmatrix}$$

one gets

$$\begin{bmatrix} P_i & (A_i + B_i K_j) \\ (*) & P_j^{-1} \end{bmatrix} \succeq 0, \forall i,j = \overline{1,s}$$

For each i, multiply the corresponding by $\alpha_i(k)$, and sum to get, $\forall i,j = \overline{1,s}$

$$\begin{bmatrix} \sum_{i=1}^{s} \alpha_i(k) P_i & (*) \\ \sum_{i=1}^{s} \alpha_i(k) (A_i + B_i K_j)^T & P_j^{-1} \end{bmatrix} \succeq 0$$

By using the Schur complement, one obtains, $\forall j = \overline{1,s}$

$$\begin{aligned} & P_j^{-1} \succeq \\ & (A(k) + B(k) K_j)^T \left(\sum_{i=1}^{s} \alpha_i(k) P_i \right)^{-1} (A(k) + B(k) K_j) \end{aligned}$$

It follows that

$$\begin{aligned} & v_j(k)^T P_j^{-1} v_j(k) \geq \\ & \quad v_j(k+1)^T \left(\sum_{i=1}^{s} \alpha_i(k) P_i \right)^{-1} v_j(k+1) \end{aligned} \tag{60}$$

Hence, if $v_j(k) \in \mathcal{E}(P_j)$ then $v_j(k+1) \in \mathrm{Co}_{i=1}^{s} (\mathcal{E}(P_i))$. Therefore $\mathrm{Co}_{i=1}^{s} (\mathcal{E}(P_i))$ is robustly controlled invariant.

It remains to show that $\mathrm{Co}_{i=1}^{s} (\mathcal{E}(P_i))$ is constraint-admissible. Using the proof of Theorem 1, it follows that (14), (15) provides a necessary and sufficient condition for satisfying the state constraints (5). For the input constraints, using (56), one has $u \in U$ if and only if one has $u \in \mathcal{U}$ if and only if

$$K_j v_j \in \mathcal{U}, \forall v_j \in \mathcal{E}(P_j), \forall j = \overline{1,s}$$

Using [17], this condition is satisfied if and only if $\exists G_j \in \mathbb{S}^{c_u}$ such that the following condition is satisfied

$$\begin{cases} \begin{bmatrix} G_j & F_u K_j P_j \\ (*) & P_j \end{bmatrix} \succeq 0 \\ G_j(l) \leq g_u(l)^2, \forall l = \overline{1,c_u} \end{cases} \tag{61}$$

where $G_j(l)$ is the lth diagonal element of G_j.

By substituting $Y_j = K_j P_j, \forall j = \overline{1,s}$ in (61), one obtains (16), (17). The proof is complete. □

C. Proof of Theorem 3: Recursive feasibility and asymptotic stability are proved at the same time. Consider the following candidate Lyapunov function, $\forall x(k) \in \mathrm{Co}_{i=1}^{s} (\mathcal{E}(P_i))$

$$V(x(k)) = \sum_{i=1}^{s} \lambda_i^*(k) \tag{62}$$

Clearly, $V(k) \geq 0, \forall k$, and $V(x(k)) = 0$ if and only if $x = 0$.

Substituting (21), (22) into (11), one gets

$$\begin{aligned} x(k+1) & = \sum_{i=1}^{s} \lambda_i^*(k) (A(k) + B(k) K_i) v_i^*(k) \\ & = \sum_{i=1}^{s} \lambda_i^*(k) v_i(k+1) \end{aligned} \tag{63}$$

where $v_i(k+1) = (A(k) + B(k) K_i) v_i^*(k)$. As shown in the proof of Theorem 2, if $v_i^*(k) \in \mathcal{E}(P_i)$, then $v_i(k+1) \in \mathrm{Co}_{j=1}^{s} (\mathcal{E}(P_j))$. Hence we can decompose $v_i(k+1)$ as

$$v_i(k+1) = \sum_{j=1}^{s} \lambda_{ij} v_{ij}(k+1) \tag{64}$$

where $\sum_{j=1}^{s} \lambda_{ij} = 1, \lambda_{ij} \geq 0$, and $v_{ij}(k+1) \in \mathcal{E}(P_j), \forall j = \overline{1,s}$.

Substituting (64) into (63), one obtains

$$\begin{aligned} & x(k+1) = \sum_{i=1}^{s} \lambda_i^*(k) \sum_{j=1}^{s} \lambda_{ij} v_{ij}(k+1) \\ & = \sum_{j=1}^{s} \sum_{i=1}^{s} \lambda_i^*(k) \lambda_{ij} v_{ij}(k+1) \\ & = \sum_{j=1}^{s} \eta_j(k) \left(\sum_{i=1}^{s} \frac{\lambda_i^*(k) \lambda_{ij}}{\eta_j} v_{ij}(k+1) \right) = \sum_{j=1}^{s} \eta_j(k) \xi_j(k+1) \end{aligned}$$

where $\eta_j(k) = \sum_{i=1}^{s} \lambda_i^*(k) \lambda_{ij}$, and $\xi_j(k+1) = \sum_{i=1}^{s} \frac{\lambda_i^*(k) \lambda_{ij}}{\eta_j} v_{ij}(k+1)$. Using the facts that $\sum_{i=1}^{s} \frac{\lambda_i^*(k) \lambda_{ij}}{\eta_j} = 1$ and $v_{ij}(k+1) \in \mathcal{E}(P_j), \forall i,j = \overline{1,s}$, one gets $\xi_j(k+1) \in \mathcal{E}(P_j), \forall j = \overline{1,s}$. This implies that $x(k+1)$ can be written as a convex combination of $\xi_j(k+1)$ with $\eta_j(k)$ being interpolating coefficients. Hence recursive feasibility is guaranteed.

One has

$$\sum_{j=1}^{s} \eta_j(k) = \sum_{j=1}^{s} \sum_{i=1}^{s} \lambda_i^*(k) \lambda_{ij} = \sum_{i=1}^{s} \lambda_i^*(k) \sum_{j=1}^{s} \lambda_{ij}$$

Since $\sum_{j=1}^{s} \lambda_{ij} = 1$, one obtains $\sum_{j=1}^{s} \eta_j(k) = \sum_{i=1}^{s} \lambda_i^*(k)$. By optimizing (63) at time $k+1$, one has $V(x(k+1)) = \sum_{i=1}^{s} \lambda_i^*(k+1) \leq \sum_{i=1}^{s} \eta_i(k)$. It follows that $V(x(k+1)) \leq \sum_{i=1}^{s} \lambda_i^*(k)$, or equivalently $V(x(k+1)) \leq V(x(k))$. Hence $V(x(k))$ is a non-increasing function.

Since the LMIs (13) are strict, it follows that there exists no initial condition $x(0) \in \mathrm{Fr}\,(\mathrm{Co}_{i=1}^{s} (\mathcal{E}(P_i)))$ such that $x(k) \in \mathrm{Fr}\,(\mathrm{Co}_{i=1}^{s} (\mathcal{E}(P_i)))$, $\forall k \geq 1$. Hence $V(x(k))$ is strictly decreasing, and $V(x(k))$ is a Lyapunov function. In other words, the closed-loop system is robustly asymptotically stable. □